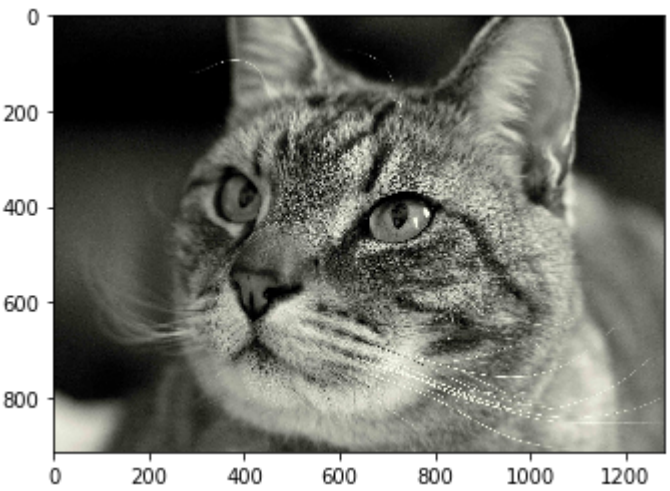
In [2]:
```
import matplotlib.pyplot as plot
import matplotlib.image as matimg
import numpy as np
import os
```

In [5]:
```
#Reading Image using matplotlib package
input_image = matimg.imread('image.jpg')
```

In [14]:
```
##Displaying grey scale image
plot.imshow(input_image)
```

Out[14]:
```
<matplotlib.image.AxesImage at 0x292e7ff0448>
```



In [6]:
```
## Converting the image to grey scale
rgb_conv=[0.2989, 0.5870, 0.1140]
grey_image = np.dot(input_image[...,:3],rgb_conv )
```

In [11]:
```
## Initializing the singular value decomposition
u,s,vt = np.linalg.svd(grey_image, full_matrices =  False)
```

In [12]:
```
## Deriving diagonal singular values of s
s_diag = np.diag(s)
s_diag.shape
```

Out[12]:
```
(913, 913)
```

In [13]:
```
## Taking a sample singular value range
k = [1,5,10,50,100,150,500]

## Constructing approximation for given k values
img_iter = 0
for i in k:
    approx = u[:,:i] @ s_diag[:i,:i] @ vt[:i, :]
    plot.figure(img_iter+1)
    img_iter = img_iter+1
    img_approx = plot.imshow(256-approx)
    img_approx.set_cmap('gray')
    plot.title('Image approximation when k='+str(i))
    plot.show()
```

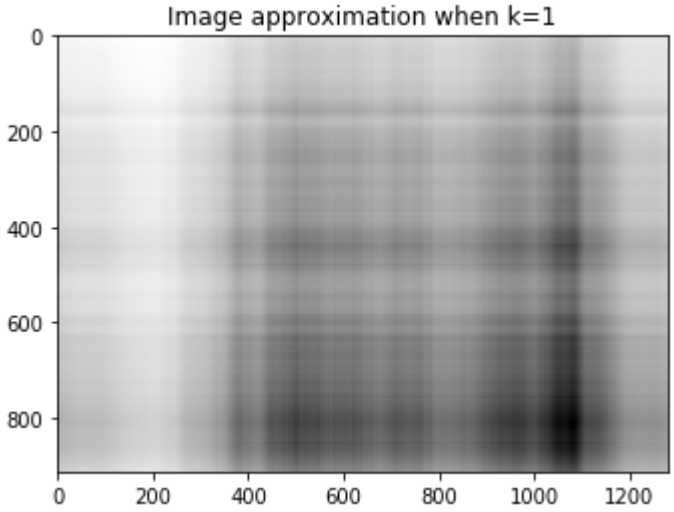


Image approximation when k=1
0
200
400
600
800
0
200
400
600
800
1000
1200

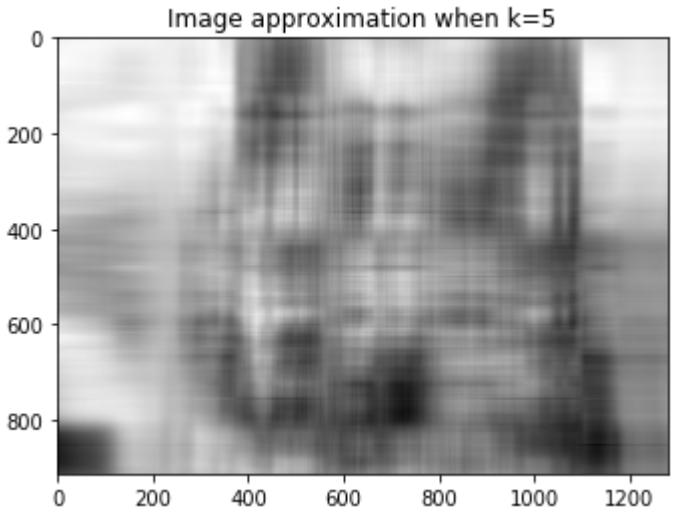


Image approximation when k=5
0
200
400
600
800
0
200
400
600
800
1000
1200

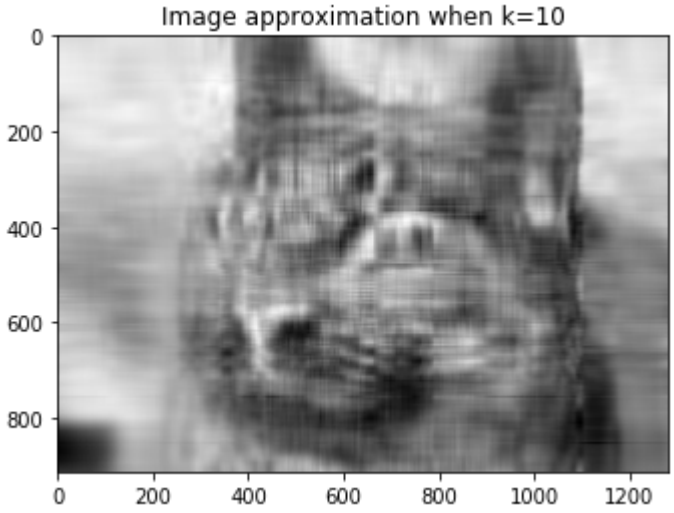


Image approximation when k=10
0
200
400
600
800
0
200
400
600
800
1000
1200

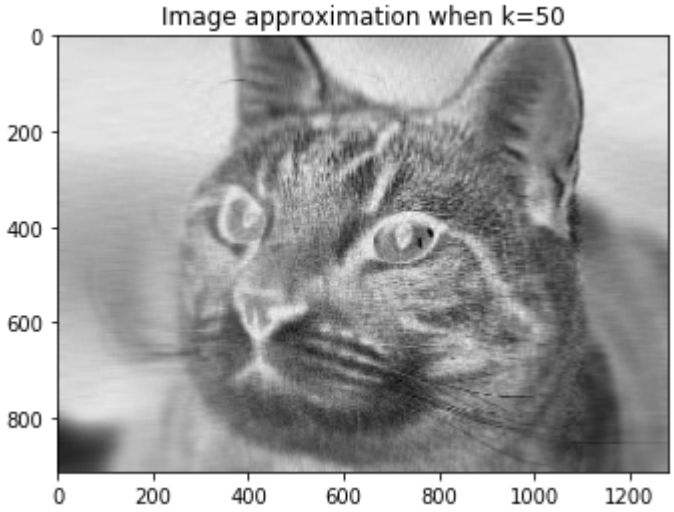


Image approximation when k=50
0
200
400
600
800
0
200
400
600
800
1000
1200

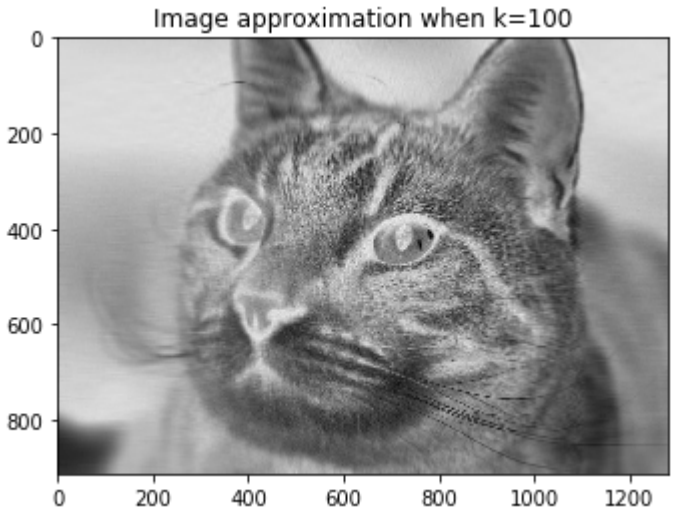


Image approximation when k=100
0
200
400
600
800
0
200
400
600
800
1000
1200

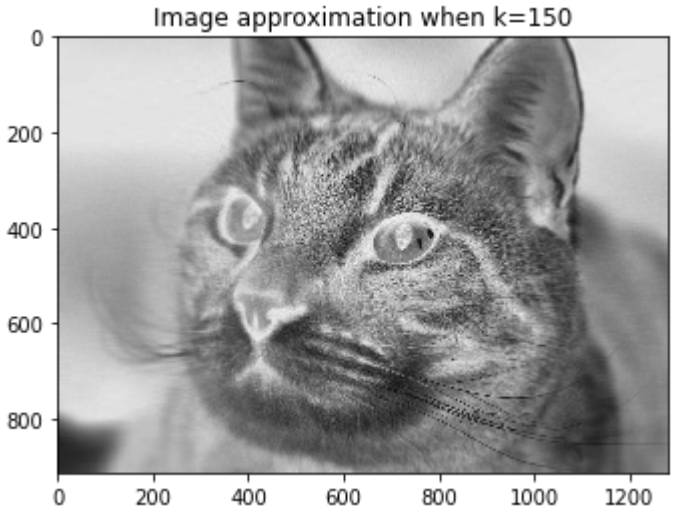
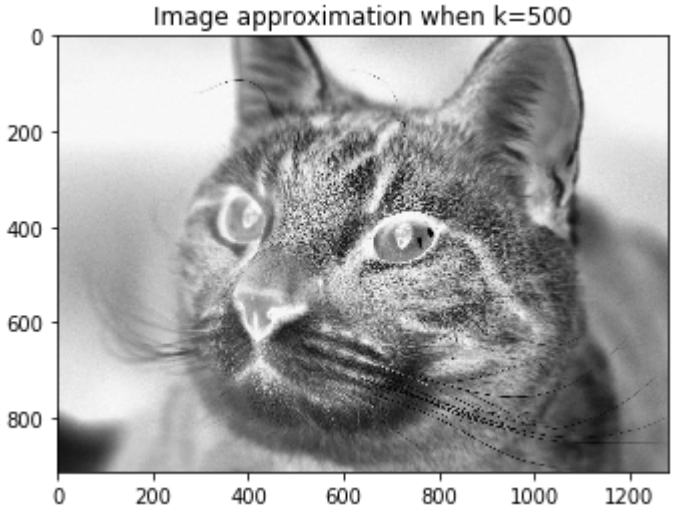


Image approximation when k=150
0
200
400
600
800
0
200
400
600
800
1000
1200



In [ ]:
```
## End of Code
```